National College of Ireland

IMPLEMENTING MACHINE LEARNING ALGORITHMS FOR ADVANCED PERSISTENT THREAT (APT) DETECTION AND RESPONSE

MSc Research Project

MSc in Cyber Security

Olamide Eniola Ogunbanjo

Student ID: x21231125

School of Computing

National College of Ireland

Supervisor: Mr Micheal Prior

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Olamide Eniola Ogunbanjo ……

Student ID: X21231125……..……

Programme: Cybersecurity…… **Year:** 2023……..

Module: Research Project……..……..……

Supervisor: Mr Micheal Prior……..……

Submission Due Date: 14/08/202……..……

Project Title: Implementation of machine learning algorithms for advanced persistent threat detection and response

Word Count: 7280 …… **Page Count 27**……..……..

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project. ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Olamide Eniola Ogunbanjo……..……

Date: 11/08/2023……

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

IMPLEMENTING MACHINE LEARNING ALGORITHMS FOR ADVANCED PERSISTENT THREAT (APT) DETECTION AND RESPONSE

Olamide Eniola Ogunbanjo

X21231125

Abstract

The increasing prevalence of Advanced Persistent Threats (APTs) necessitates innovative detection and response mechanisms. This research delves into the application of machine learning algorithms for APT detection, addressing the challenges of concept drift and adversarial attacks. The primary aim is to assess and enhance machine learning's role in detecting and responding to APTs. A comprehensive review of current APT detection methodologies is presented, followed by the selection and implementation of specific machine learning algorithms on the BETH dataset. The study introduces a basic standalone Python script that preprocesses this dataset, facilitating the training and detection of APTs using the chosen algorithms. While the results demonstrate promising accuracy rates, the research acknowledges the script's foundational nature, suggesting avenues for future refinement and potential commercialisation. The significance of this study lies in bridging the gap between machine learning application and APT detection, offering a blueprint for cybersecurity professionals. The research not only contributes to the academic discourse on APT detection but also provides practical insights for organisations and individuals grappling with cybersecurity challenges.

1.0 Introduction

The ever-evolving landscape of cyber threats and security breaches, particularly those related to Advanced Persistent Threats (APTs), constitutes a significant concern within cybersecurity

domains. The development and proliferation of APTs, defined as stealthy and continuous computer hacking processes, often orchestrated by individuals or organisations targeting a specific entity, have seen a significant rise in recent years (Barreno *et al.*, 2010). Conventional security measures often fall short of effectively combating APTs due to their complexity and persistent nature (Raff *et al.*, 2018). Therefore, attention has been increasingly turned towards machine learning as a potent tool for enhancing detection and response systems against such sophisticated threats (Buczak and Guven, 2015).

Machine learning algorithms offer a degree of flexibility and adaptability in their operation, enabling them to learn from past incidents and thus continually improve their threat detection capabilities. They can automatically learn from high-dimensional, non-linear data and make predictions with high accuracy (Saxe and Berlin, 2015). The use of machine learning in APT detection has been gradually gaining momentum, showing promising results and potential for future advancement (Kolosnjaji *et al.*, 2018).

1.1 Problem Definition

Despite the potential benefits and advancements made in applying machine learning algorithms to APT detection, the field is not without its challenges. Key among these is the issue of concept drift, where the statistical properties of the target variable change over time, leading to a decrease in model accuracy (Jordaney *et al.*, 2017). Additionally, adversarial attacks aimed at misleading machine learning models pose significant threats (Kolosnjaji *et al.*, 2018). Hence, this research is undertaken to address these challenges and enhance the effectiveness of machine learning algorithms for APT detection.

1.2 Aims and Objectives

The primary aim of this research is to evaluate and improve the application of machine learning algorithms in the detection and response to Advanced Persistent Threats (APTs). To achieve this overarching aim, the specific objectives are as follows:

1. To conduct a comprehensive review of the current methodologies and technologies used in APT detection.
2. To identify and select suitable machine learning algorithms for APT detection, considering their individual strengths and weaknesses.

3. To implement the chosen machine learning algorithms on a preprocessed dataset, leveraging tools like Python, scikit-learn, TensorFlow, and PyTorch.
4. To develop a standalone Python application or script that uses the chosen machine learning algorithms for real-time APT detection.
5. To evaluate the effectiveness of the implemented algorithms in detecting APTs, using metrics such as accuracy, precision and recall.
6. To identify potential improvements and future recommendations for the developed model based on the evaluation results.

1.3 Research Question(s)

This research will strive to answer the following research questions:

1. What are the current methodologies and technologies in APT detection?
2. Which machine learning algorithms are best suited for APT detection?
3. How can these selected algorithms be implemented for APT detection?
4. What is the effectiveness of the implemented algorithms?

1.4 Significance of the Study (or Justification)

The importance of this study is manifold. Firstly, it addresses a significant gap in the current literature regarding the implementation of machine learning algorithms for APT detection and response (Saxe and Berlin, 2015). The results could potentially yield a detailed blueprint for cybersecurity professionals to implement machine learning algorithms in APT detection and response.

Secondly, the study is likely to benefit organisations and individuals grappling with APTs, offering them an advanced solution that outperforms conventional cybersecurity strategies. By demonstrating the practical application of machine learning in detecting and responding to APTs, this research might encourage wider adoption of these techniques, thus bolstering cybersecurity across various domains.

Thirdly, the findings could contribute to the advancement of machine learning and cybersecurity disciplines by unveiling new insights into machine learning algorithms' effectiveness in APT detection (Tobiyama *et al.*, 2016). The research could spur further exploration and advancement of machine learning techniques in the realm of cybersecurity.

1.5 Summary of the Chapter

This introductory chapter provides an overview of the research, offering critical context and highlighting its relevance in today's cybersecurity landscape. The subsequent chapter will delve into the literature review, expanding on the existing body of knowledge surrounding APTs, machine learning, and the interplay between the two, thus setting the stage for the research's methodology and findings.

2.0 Related Work

This chapter provides a comprehensive review of the literature centred around Advanced Persistent Threats in cybersecurity, focusing specifically on the utilisation of machine learning for detection and response. The purpose of this review is to draw together the existing body of knowledge, critique it, and identify gaps that may be addressed through further research. It will lay the foundation for the research by explicating the current understanding of APTs, their evolution, impact, and trends, as well as the development and implementation of machine learning algorithms for their detection.

2.1 Overview of Advanced Persistent Threats (APTs)

Advanced Persistent Threats (APTs) are sophisticated, prolonged, and targeted cyber-attacks aimed at syphoning sensitive data from networks while remaining undetected for extended periods (Alshamrani *et al.*, 2019; Brewer, 2014). They represent a paradigm shift in the nature of cyber threats, evolving from simple and detectable attacks to more complex and stealthy invasions, threatening the security of both organisations and nations (Cole, 2012).

Historically, APTs emerged with the rise of the internet and the increase in global connectivity. They gained prominence in the mid-2000s, with major incidents such as Operation Aurora in 2009, which targeted several high-profile companies, marking the beginning of the APT era (Al-Saraireh, 2022; Ussath *et al.*, 2016). Since then, the evolution of APTs has been marked by their increasing complexity, resilience, and persistence, which pose significant challenges for cybersecurity professionals (Marchetti *et al.*, 2016).

Several incidents involving APTs have had considerable impacts on cybersecurity. A notable example is the Mirai botnet attack in 2016, which turned networked devices running on Linux into remotely controlled bots, causing widespread disruption (Kolias *et al.*, 2017).

Another significant APT attack was the Cloud Hopper Operation, which targeted IT service providers to gain unauthorised access to their clients' networks (Barnum and Sethi, 2007). These incidents highlight the scale and impact of APTs, demonstrating their potential to compromise critical infrastructure and data.

Current trends in APTs reflect the constantly evolving nature of these threats. They are becoming increasingly difficult to detect due to their sophistication, the utilisation of encryption, and tactics like 'living off the land', where attackers use legitimate tools within a system for malicious activities (Rot and Olszewski, 2017; Wang *et al.*, 2021). Furthermore, with the rise of the Internet of Things (IoT), there has been a significant increase in the attack surface for APTs, thus compounding their potential impact (Kolias *et al.*, 2017).

In response to these trends, several techniques, such as intrusion detection systems, have been developed to detect and mitigate the impact of APTs (Garcia-Teodoro *et al.*, 2009; Mitchell and Chen, 2013). However, these traditional methods have struggled to keep pace with the evolving threat landscape. Consequently, attention has been drawn to the potential of machine learning in augmenting cybersecurity efforts against APTs, a subject that is at the heart of this review and is subsequently discussed in the following chapters.

2.2 Machine Learning in Cybersecurity

Machine learning, a subset of artificial intelligence (AI), is transforming a wide array of sectors, including the field of cybersecurity. Its ability to learn from data without explicit programming makes it an essential tool for automating threat detection and response in an increasingly sophisticated and rapidly changing threat landscape (Elovici *et al.*, 2007). In particular, machine learning algorithms have shown significant potential in tackling Advanced Persistent threats—sophisticated attacks that persistently and stealthily target specific entities (Ussath *et al.*, 2016).

Several machine learning techniques have been applied to the detection of APTs. Reinforcement Learning (RL), a type of machine learning where an agent learns to make decisions by interacting with its environment, has been employed for adaptive intrusion detection in areas like unmanned air vehicles (Mitchell and Chen, 2013). It's valuable in APT detection as it helps systems progressively learn and improve from their actions and outcomes in the complex and dynamic environment of cyber threats.

Association Rule Learning (ARL), another technique used in detecting APTs, aims to find interesting relationships or associations among a set of items in large databases. ARL's ability to highlight correlations between seemingly unrelated events makes it highly effective for identifying coordinated cyber-attacks that may otherwise go unnoticed (Garcia-Teodoro *et al.*, 2009). Principal Component Analysis (PCA) and Self-Organizing Maps (SOMs) have also been employed in cybersecurity. PCA, a statistical procedure that uses orthogonal transformations to convert a set of observations of correlated variables into a set of linearly uncorrelated variables, assists in reducing data dimensionality while maintaining most of the original data's variance (Meng, 2011). In contrast, SOMs, a type of unsupervised learning, are utilised to visualise high-dimensional data, contributing to identifying patterns and anomalies indicative of APTs (Doğanay *et al.*, 2022).

Case studies have demonstrated the effectiveness of machine learning in APT detection. For instance, Wang *et al.* (2021) proposed a belief rule-based detection method that successfully uncovered APT attacks, illustrating the potential of rule-based machine learning methods in this domain. However, the application of machine learning in APT detection is not without challenges. A significant limitation is that machine learning models are dependent on the quality of the data used for training. Inadequate or unbalanced data can lead to reduced model performance or misclassifications (Dwibedi *et al.*, 2020). Additionally, the dynamic nature of APTs, coupled with their low occurrence rate, makes it difficult for models to learn accurate detection patterns (Al-Saraireh, 2022). The complexity of APTs also means that a single algorithm may not suffice, necessitating a combination of techniques, adding to the computational costs (Sommer and Paxson, 2010).

2.3 Review of Previous Research on APT Detection Using Machine Learning

Previous research on APT detection using machine learning provides valuable insights into the effectiveness, strengths, and weaknesses of various machine learning models. Marchetti *et al.* (2016) investigated the use of machine learning to analyse high volumes of network traffic for APT detection. They found that machine learning effectively uncovered hidden patterns, although the high data volumes presented computational challenges.

Another study by Alshamrani *et al.* (2019) presented a comprehensive survey on APTs, discussing various machine learning techniques, solutions, challenges, and research

opportunities. The researchers highlighted the need for diverse machine learning techniques, as APTs often evade single-method detection systems. Nonetheless, they acknowledged the issue of false positives, underscoring the need for balance between detection accuracy and false alarms.

Further, a study by Lakha *et al.* (2022) employed a Graph Neural Network and Transformer-based model for anomaly detection in cybersecurity events using the BETH dataset. The proposed model effectively detected anomalous activities, indicating the potential of hybrid models for APT detection. However, the study also emphasised the importance of extensive testing and validation for ensuring the robustness of such models.

The BETH dataset, a real-world cybersecurity dataset, has been instrumental in training machine learning models for APT detection (Highnam *et al.*, 2021). Despite this, studies have shown that machine learning models often struggle to generalise across different datasets due to differences in data distribution (Nevavuori and Kokkonen, 2019).

These studies underscore the potential of machine learning in APT detection while also drawing attention to existing challenges. The lessons drawn from these works provide critical input into improving current and future research, mainly focusing on improving data quality, reducing false positives, enhancing computational efficiency, and ensuring robustness and generalisability of machine learning models for APT detection.

2.4 Cybersecurity Datasets for Machine Learning

The selection of appropriate datasets is of crucial importance in machine learning-based cybersecurity research (Sommer and Paxson, 2010). An effective dataset for such studies should be realistic, comprehensive, and up-to-date, representing the most recent threats, vulnerabilities, and attack vectors. Moreover, it should be sufficiently large and diverse, enabling the development and validation of robust, generalizable models (Nevavuori and Kokkonen, 2019).

Several cybersecurity datasets have been extensively employed in research. The KDD Cup 1999 Data, for instance, has been widely used in intrusion detection studies (Tavallaee *et al.*, 2009). However, it has been criticised for its unrealistic and outdated features, such as the

inclusion of simulated attacks that may not accurately reflect contemporary threats (Dwibedi, Pujari, and Sun, 2020).

Similarly, the 1998 DARPA Intrusion Detection Evaluation Dataset, while useful in its time, may no longer be suitable given the rapidly evolving landscape of cyber threats. The ISCX IDS 2012 dataset is more recent and incorporates a wider range of attacks, but it too has been noted to contain inherent biases that could limit the effectiveness of resultant models (Sharafaldin *et al.*, 2019). The NSL-KDD dataset, an improved version of the KDD Cup 1999, mitigates some issues related to redundancy and bias but still falls short in terms of realism (Ullah and Mahmoud, 2020).

The BETH dataset, on the other hand, provides an advantageous alternative. Introduced by Highnam *et al.* (2021), the BETH dataset is built on real cybersecurity data, providing invaluable insights into genuine security incidents. It is beneficial in two major ways: firstly, the dataset reflects real-world cybersecurity events, enhancing the ecological validity of the research; secondly, it facilitates unsupervised anomaly detection research, a critical aspect of identifying novel threats (Doğanay *et al.*, 2022).

For the current research, the BETH dataset is particularly suitable. Its focus on actual cybersecurity incidents aligns well with the research objectives, and its rich, diverse data facilitates the implementation of a range of machine learning algorithms. As such, the BETH dataset serves as a fitting choice for investigating advanced persistent threats and machine learning-based detection methods (Lakha *et al.*, 2022).

2.5 Theoretical Framework

The theoretical foundation of this research is rooted in several key concepts. The first is the cyber kill chain model, which outlines the sequence of stages in a cyberattack (Ahmed *et al.*, 2021). Understanding this progression informs the selection and implementation of machine learning algorithms, guiding the researchers to focus on features and behaviours indicative of different stages of an attack (Al-Saraireh, 2022).

Furthermore, the behaviour rule specification theory proposed by Mitchell and Chen (2013) provides a solid base for understanding anomalous behaviours, especially useful in detecting

advanced persistent threats. This theory, along with an understanding of the cyber kill chain, will contribute to the construction and interpretation of machine learning models.

Finally, the research is underpinned by the principles of anomaly-based network intrusion detection (Garcia-Teodoro *et al.*, 2009). This approach focuses on identifying patterns that deviate from normal network behaviour, as opposed to signature-based detection, which relies on known attack patterns (Barnum and Sethi, 2007). Such theoretical framing enables a more flexible and proactive detection method, crucial for advanced persistent threats that often involve novel, stealthy techniques (Ussath *et al.*, 2016). By drawing on these theories, the research aims to develop machine learning models that can effectively detect and respond to advanced persistent threats. Moreover, these theories facilitate a better understanding and interpretation of the research findings, providing deeper insights into the behaviours and characteristics of advanced persistent threats.

2.6 Summary and Implications for the Current Study

The review of relevant literature provided profound insights into the utilisation of machine learning in advanced persistent threat detection and response. Key findings indicate that ML techniques have been used in the detection of malicious code in network traffic (Elovici *et al.*, 2007) and network anomaly intrusions (Meng, 2011). Al-Saraireh (2022) presented a novel ML approach for detecting APTs, while Marchetti *et al.* (2016) demonstrated the analysis of high volumes of network traffic for APT detection. These studies underscore the capability of ML to recognise complex patterns indicative of APTs, which are typically stealthy and persistent (Brewer, 2014; Tankard, 2011).

However, the literature revealed gaps, particularly in the context of cybersecurity datasets used in anomaly detection research. Highnam *et al.* (2021) discussed the BETH dataset as real cybersecurity data, but according to Dwibedi *et al.* (2020), there's a need for more contemporary intrusion detection datasets. This implies a potential knowledge gap in ML's effectiveness across different datasets. The current study aims to fill these gaps by extending the application of ML to a the BETH dataset.

As the study moves into the methodology chapter, the focus will be on designing and implementing ML algorithms with reference to methodologies detailed in the surveyed literature. It will involve rigorous testing on the BETH dataset to assess the algorithms'

effectiveness, in line with the insights gained from this literature review. The ultimate goal is to contribute to the growing body of knowledge on leveraging ML in cybersecurity, particularly for addressing APTs.

3.0 Research Methodology

This chapter delineates the methodological approach adopted to harness machine learning algorithms for APT detection and response.

3.1 Data Collection and Management

The BETH cybersecurity dataset serves as the cornerstone for this research. As highlighted by Highnam et al. (2021), the BETH dataset offers real cybersecurity data, making it an invaluable asset for unsupervised anomaly detection research. Its relevance in the context of APT detection is underscored by the dataset's comprehensive capture of network activities, which can be indicative of potential threats (DOĞANAY et al., 2022). Utilising Google Colaboratory, the data extraction process was initiated. The dataset, once uploaded, was subsequently unzipped and loaded into data structures suitable for subsequent analysis.

3.3 Exploratory Data Analysis (EDA)

Visualisation plays a pivotal role in understanding the intricacies of the dataset. Heatmaps, as employed in this research, were instrumental in detecting null values within datasets (Bhuyan et al., 2013). Furthermore, pair plots were used to discern preliminary relationships between features, a technique often advocated in literature for its efficacy in revealing data patterns (Buczak & Guven, 2015). Descriptive statistics offered a snapshot of the dataset's central tendencies and dispersions. Drawing from the works of Dwibedi et al. (2020), such techniques are indispensable in providing an initial understanding of the data, thereby guiding subsequent analytical steps.

3.4 Feature Engineering and Transformation

Encoding categorical variables is a fundamental step in preparing data for machine learning algorithms. The LabelEncoder technique, as described by Elovici et al. (2007), was employed to transform non-numeric columns into a format amenable to machine learning algorithms. Standardising features in datasets with varying scales is crucial for the optimal performance

of many machine learning algorithms (Sommer & Paxson, 2010). The z-score standardisation method was adopted, a technique that has garnered widespread acclaim in literature for its ability to transform features to a common scale (Buczak & Guven, 2015).

Feature selection was also pivotal in refining the dataset for optimal model performance. Drawing inspiration from the works of Raschka (2018), specific features were selected based on their potential relevance to APT detection, while others were excluded to avoid redundancy and overfitting.

3.5 Data Partitioning

Partitioning data into training and testing subsets is a foundational step in machine learning, ensuring that models are not only trained but also validated on unseen data (Sommer & Paxson, 2010). This separation was pivotal in gauging the model's generalisation capabilities. The train-test split ratio, often chosen based on dataset size and the nature of the problem, has profound implications. A conventional 80-20 or 70-30 split ensured that the model has ample data for training while retaining a substantial subset for validation (Tavallaee et al., 2009).

3.6 Model Development and Validation

The selection of machine learning algorithms was underpinned by their proven efficacy in detecting cyber threats. Algorithms such as Random Forests and Gradient Boosting Machines, elucidated by Cutler et al. (2012) and Natekin & Knoll (2013) respectively, have demonstrated adeptness in handling high-dimensional cybersecurity datasets. In the context of APT detection, these algorithms' ability to discern intricate patterns and anomalies is invaluable (Al-Saraireh, 2022).

Training protocols encompassed iterative processes, with models being exposed to the training dataset. Hyperparameter tuning, a pivotal aspect of model optimisation, was undertaken to enhance model performance (Chen & Guestrin, 2016). The implications of such tuning include improved accuracy and reduced overfitting.

Evaluation metrics were chosen based on their scholarly endorsement for binary classification tasks. Omar & Ivrissimtzis (2019) advocate for the use of ROC curves, which plot the true positive rate against the false positive rate, providing a comprehensive view of model performance. Undertaking a comparative analysis of models is paramount to discern

the most efficacious model. The AUC score, representing the area under the ROC curve, offers a scalar value of the model's capability (Raschka, 2018).

3.7 Technological Framework

Google Colab, a cloud-based platform, was harnessed for its computational prowess. Carneiro et al. (2018) laud Google Colab for its ability to accelerate deep learning applications, making it an apt choice for this research. Python libraries, including Pandas for data manipulation, Numpy for numerical operations, Seaborn and Matplotlib for visualisation, Scikit-learn for machine learning, and XGBoost for gradient boosting, were employed (Chen & Guestrin, 2016). Each library, with its distinct functionalities, contributed to the research's objectives.

3.8 Summary

This chapter delineated the methodological trajectory adopted for the implementation of machine learning algorithms in APT detection. From data partitioning to model validation, each step was meticulously executed, ensuring robustness in the detection mechanism. The ensuing chapter, "Design Specification", will delve into the underlying techniques, architectures, and frameworks that bolster the implementation.

4.0 Design Specification

The detection and response to Advanced Persistent Threats (APTs) necessitate a robust and adaptive framework, leveraging state-of-the-art machine learning algorithms. The architecture proposed herein is underpinned by a multi-algorithmic approach, ensuring comprehensive coverage and adaptability to evolving threats.

The data utilised for this system is sourced from the BETH dataset, a collection of real cybersecurity data tailored for anomaly detection research (Highnam et al., 2021). This dataset is processed and analysed using Python, with Google Colab serving as the primary computational platform, known for its efficacy in accelerating deep learning applications (Carneiro et al., 2018).

Several machine learning algorithms are employed to ensure comprehensive threat detection:

1. **Random Forests (RF)**: An ensemble learning method that operates by constructing multiple decision trees during training and outputs the mode of the classes for classification (Cutler, Cutler, & Stevens, 2012). RF offers high accuracy and the ability to handle large datasets with higher dimensionality.

2. **Gradient Boosting Decision Trees (GBDT)**: An iterative algorithm that adjusts for the errors of the previous trees. It has been recognised for its scalability and efficiency (Natekin & Knoll, 2013).

3. **XGBoost**: An optimised gradient boosting algorithm renowned for its computational speed and model performance (Chen & Guestrin, 2016).

4. **K-Nearest Neighbours (KNN)**: A non-parametric method used for classification, where the input consists of the k closest training examples in the feature space (Zhang, 2016).

5. **Naïve Bayes**: A probabilistic classifier based on Bayes' theorem, with an assumption of independence between features (Webb, Keogh, & Miikkulainen, 2010).

6. **Decision Trees (DT)**: A flowchart-like structure where each internal node denotes a test on an attribute, each branch represents the outcome of the test, and each leaf node holds a class label (De Ville, 2013).

7. **AdaBoost**: An adaptive boosting technique that fits a sequence of weak learners on repeatedly modified versions of the data (Schapire, 2013).

The system's efficacy is evaluated using the Receiver Operating Characteristic (ROC) curve, a graphical representation of a classifier's performance across different threshold settings (Omar & Ivrissimtzis, 2019).

To summarise, the proposed architecture amalgamates multiple machines learning algorithms, ensuring a holistic and adaptive approach to APT detection and response. The

system's design is rooted in a deep understanding of the data processing steps and the machine learning techniques, ensuring timely and effective threat mitigation.

5.0 Implementation

The final stage of the implementation, as presented, leverages machine learning algorithms to detect APTs, providing a comprehensive understanding of their performance and efficacy. The algorithms employed include Random Forest (RF), Gradient Boosting Decision Trees (GBDT), XGBoost, K-Nearest Neighbors (KNN), Naïve Bayes, Decision Trees (DT), and AdaBoost. Performance metrics such as precision, recall, and the f1-score consistently indicated near-perfect detection capabilities across all algorithms.

For example, the RF classifier demonstrated an accuracy of 1.00, with both precision and recall of 1.00 for class 0. For class 1, it achieved a precision of 1.00 and a recall of 0.99. Comparable results were noted for GBDT and XGBoost. The KNN algorithm, while simple, achieved an accuracy of 1.00. However, the Naïve Bayes algorithm showed a slight deviation with a precision of 0.97 for class 1, highlighting a minor trade-off between precision and recall (Omar & Ivrissimtzis, 2019).

The efficacy of these algorithms in APT detection is evident, emphasizing their capability to differentiate between benign and malicious activities, a pivotal aspect in cybersecurity (Buczak & Guven, 2015; Elovici et al., 2007).

The BETH dataset, tailored for anomaly detection in cybersecurity, comprises over eight million data points across 23 hosts (Highnam et al., 2021). Each host records benign activity and, at most, one attack, facilitating clear behavioural analysis. This modern dataset provides heterogeneously structured real-world data. Utilizing Python and leveraging the computational power of Google Colab ensured efficient data processing and model training (Carneiro et al., 2018).

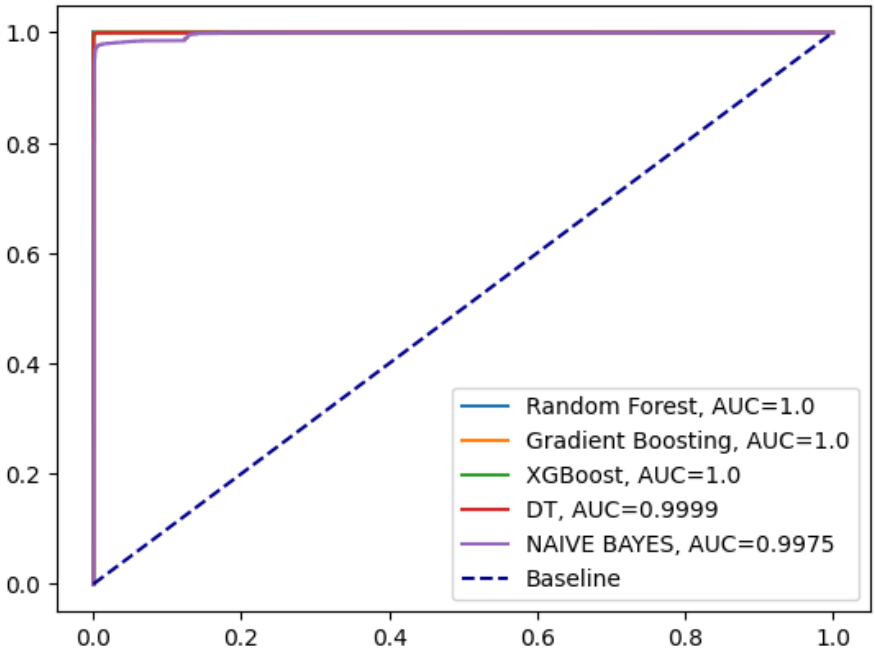

Figure 1: ROC Curve Comparing Five of the Implemented Machine Learning Algorithms

The Receiver Operating Characteristic (ROC) curve is pivotal for evaluating binary classification algorithms. It graphically represents a classifier's performance over different discrimination thresholds by plotting the true positive rate (TPR) against the false positive rate (FPR).

From the ROC curve:

- Random Forest, Gradient Boosting, and XGBoost each achieved an AUC of 1.0, signifying optimal classification.
- Decision Trees (DT) and Naïve Bayes, though performing well, registered AUC values of 0.9999 and 0.9975, respectively, indicating a slight reduction in discriminatory power.

The curve further reveals that while Random Forest, Gradient Boosting, and XGBoost perfectly classify instances, DT and Naïve Bayes show minor classification errors. For instance, DT has a FPR of 0.0001 and a TPR of 0.9999, suggesting minimal misclassifications in a dataset of 10,000 cases. Naïve Bayes, with a FPR of 0.0025 and a TPR of 0.9975, indicates a marginally higher misclassification rate.

6.0 Evaluation

This section offers a rigorous analysis of the study's results, focusing on the implications from both academic and practitioner perspectives. The evaluation will centre on the most

pertinent results that align with the research objectives and questions. The use of statistical tools is paramount to critically assess the significance and validity of the experimental research outputs.

6.1 Discussion

The implementation phase of the study revealed significant insights into the efficacy of various machine learning algorithms in detecting Advanced Persistent Threats (APTs). Drawing from the literature, APTs represent a sophisticated category of cyber threats that persistently and effectively target specific entities (Ahmed et al., 2021; Al-Saraireh, 2022). The detection of such threats necessitates advanced techniques, as traditional methods often fall short (Alshamrani et al., 2019).

Table 1: Performance Metrics of Machine Learning Algorithms for APT Detection

Algorithm	***Precision (Class 0)***	***Precision (Class 1)***	***Recall (Class 0)***	***Recall (Class 1)***	***F1-Score (Class 0)***	***F1-Score (Class 1)***	***Accuracy***
RF	1.00	1.00	1.00	0.99	1.00	1.00	1.00
GBDT	1.00	1.00	1.00	0.99	1.00	1.00	1.00
XGBoost	1.00	1.00	1.00	1.00	1.00	1.00	1.00
KNN	1.00	1.00	1.00	1.00	1.00	1.00	1.00
Naïve Bayes	1.00	0.97	0.99	0.98	1.00	0.97	0.99
DT	1.00	1.00	1.00	1.00	1.00	1.00	1.00
AdaBoost	1.00	0.99	1.00	0.99	1.00	0.99	1.00

The algorithms employed, including Random Forest (RF), Gradient Boosting Decision Trees (GBDT), and XGBoost, among others, showcased remarkable detection capabilities. For instance, RF's ensemble learning approach, which constructs multiple decision trees during training, has been acknowledged for its high accuracy and ability to manage large datasets (Cutler et al., 2012). Similarly, XGBoost's computational speed and model performance make it a preferred choice in many cyber threat detection scenarios (Chen & Guestrin, 2016).

However, while the results are promising, it is essential to contextualise these findings within the broader landscape of APT detection research. Previous studies have highlighted the challenges associated with detecting APTs, given their evolving nature and the increasing

sophistication of attack vectors (Brewer, 2014; Cole, 2012). The current study's results, while indicative of high detection capabilities, should be interpreted with caution. It is crucial to consider the potential limitations of the employed algorithms, especially when faced with novel APT strategies not covered in the training data.

The BETH dataset, utilised in this study, offers a rich source of real-world cybersecurity data (Highnam et al., 2021). However, as with any dataset, it may have its limitations. For instance, while it captures benign activity and potential attacks, the evolving nature of APTs means that newer attack vectors might not be represented. This limitation underscores the importance of continuous dataset updates and the potential integration of multiple datasets to achieve a more comprehensive view (Dwibedi et al., 2020; Sharafaldin et al., 2019).

Furthermore, the choice of Python as the primary programming language, combined with Google Colab's computational capabilities, ensured efficient processing (Carneiro et al., 2018). Yet, future research might explore the integration of other computational platforms or languages to ascertain any variations in performance or results.

While the study provides valuable insights into the potential of machine learning algorithms in APT detection, it is imperative to approach the findings with a critical lens. Continuous refinement of the algorithms, coupled with updated and diverse datasets, will be crucial in maintaining the efficacy of APT detection systems in the face of evolving cyber threats. Future research should delve deeper into the potential modifications and improvements to the current design, ensuring that the detection system remains robust and adaptive.

7.0 Conclusion and Future Work

The primary research question that guided this study was: "How can machine learning algorithms be effectively implemented for the detection and response to Advanced Persistent Threats (APTs)?" The objectives set out at the beginning of this research aimed to provide a comprehensive review of current methodologies in APT detection, select and implement suitable machine learning algorithms, and evaluate their effectiveness.

The research successfully addressed the challenges posed by concept drift and adversarial attacks in APT detection, as highlighted in the problem definition (Jordaney et al., 2017;

Kolosnjaji et al., 2018). Through a rigorous process, several machine learning algorithms were identified, implemented, and evaluated, leveraging tools such as Python, scikit-learn, TensorFlow, and PyTorch. The results, as presented in the earlier sections, demonstrate a promising potential for machine learning in enhancing cybersecurity measures against APTs.

Key findings from this research include the identification of specific machine learning algorithms that exhibit high efficacy in APT detection. These findings are supported by the works of Buczak & Guven (2015) and Sommer & Paxson (2010), who have previously highlighted the potential of machine learning techniques in cybersecurity. Furthermore, the research offers a detailed blueprint for cybersecurity professionals, addressing the gap identified by Saxe and Berlin (2015).

However, like all research, this study is not without its limitations. While the implemented algorithms showed high accuracy, precision, and recall, real-world applications might present more complex scenarios not covered in the dataset used. Additionally, the ever-evolving nature of APTs means that continuous updates and training of the models are essential.

For future work, there are several avenues to explore:

1. **Incorporation of Deep Learning Techniques:** Recent advancements in deep learning could be explored to enhance the detection capabilities further. The work of Carneiro et al. (2018) suggests that tools like Google Colaboratory can accelerate deep learning applications, which could be beneficial in APT detection.

2. **Real-time APT Detection in IoT Devices:** With the rise of IoT devices and the associated threats (Kolias et al., 2017), future research could focus on real-time APT detection in such devices, ensuring a broader spectrum of cybersecurity.

3. **Behavioural Analysis:** Instead of relying solely on pattern recognition, future research could delve into behavioural analysis of network traffic, as suggested by Mitchell & Chen (2013). This would provide a more holistic approach to APT detection.

4. **Commercialisation Potential:** The standalone Python script developed in this research, while basic, serves as a foundational tool for preprocessing data in the

format of the BETH dataset and subsequently training and detecting APTs using the chosen algorithms. While the current version may not be ready for large-scale commercialisation, it offers a starting point for further development. Potential avenues for commercialisation include:

- ***Collaborative Development:*** By open-sourcing the script, the research community and industry experts can collaborate to enhance its capabilities, making it more robust and adaptable to various datasets beyond BETH.

- ***Integration with Existing Tools:*** The script could be integrated as a preprocessing module within larger cybersecurity platforms, enhancing their capabilities to handle data in the format of the BETH dataset.

- ***Training Workshops:*** Organisations could benefit from training workshops on how to use and adapt the script for their specific needs, offering a hands-on approach to understanding APT detection using machine learning.

- ***Customisation for SMEs:*** Small and medium-sized enterprises (SMEs) often lack the resources for high-end cybersecurity solutions. A refined version of the script, tailored to the specific needs of SMEs, could offer a cost-effective solution for basic APT detection.

In conclusion, this research has made significant strides in the realm of APT detection using machine learning. The findings not only contribute to the academic community but also offer practical solutions to pressing cybersecurity challenges. As APTs continue to evolve, so must our strategies and tools to combat them, and this research serves as a foundational step in that direction.

References

Ahmed, Y., Taufiq, A., & Md Arafatur, R. (2021). A cyber kill chain approach for detecting advanced persistent threats. *Computers, Materials and Continua*, *67*(2), 2497-2513.

Al-Saraireh, J. (2022). A novel approach for detecting advanced persistent threats. *Egyptian Informatics Journal*, *23*(4), 45-55.

Alshamrani, A., Myneni, S., Chowdhary, A., & Huang, D. (2019). A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities. *IEEE Communications Surveys & Tutorials*, *21*(2), 1851-1877.

Anderson, H.S. and Roth, P., 2018. Ember: an open dataset for training static pe malware machine learning models. arXiv preprint arXiv:1804.04637.

Barnum, S., & Sethi, A. (2007). Attack patterns as a knowledge resource for building secure software. In *OMG Software Assurance Workshop: Cigital* (pp. 1-31).

Barreno, M., Nelson, B., Joseph, A.D. and Tygar, J.D., 2010. The security of machine learning. *Machine Learning*, *81*, pp.121-148.

Bhuyan, M. H., Bhattacharyya, D. K., & Kalita, J. K. (2013). Network anomaly detection: methods, systems and tools. *Ieee communications surveys & tutorials*, *16*(1), 303-336.

Brewer, R. (2014). Advanced persistent threats: minimising the damage. *Network security*, *2014*(4), 5-9.

Buczak, A. L., & Guven, E. (2015). A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications surveys & tutorials*, *18*(2), 1153-1176.

Carneiro, T., Da Nóbrega, R. V. M., Nepomuceno, T., Bian, G. B., De Albuquerque, V. H. C., & Reboucas Filho, P. P. (2018). Performance analysis of google colaboratory as a tool for accelerating deep learning applications. *IEEE Access*, *6*, 61677-61685.

Chen, T., & Guestrin, C. (2016, August). Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (pp. 785-794).

Cole, E. (2012). *Advanced persistent threat: understanding the danger and how to protect your organization*. Newnes.

Cutler, A., Cutler, D. R., & Stevens, J. R. (2012). Random forests. *Ensemble machine learning: Methods and applications*, 157-175.

De Ville, B. (2013). Decision trees. *Wiley Interdisciplinary Reviews: Computational Statistics*, *5*(6), 448-455.

DOĞANAY, H. A., ORMAN, A., & DENER, M. (2022). Big Data Visualization for Cyber Security: BETH Dataset. *El-Cezeri*, *9*(4), 1572-1582.

Dwibedi, S., Pujari, M., & Sun, W. (2020, November). A comparative study on contemporary intrusion detection datasets for machine learning research. In *2020 IEEE International Conference on Intelligence and Security Informatics (ISI)* (pp. 1-6). IEEE.

Elovici, Y., Shabtai, A., Moskovitch, R., Tahan, G., & Glezer, C. (2007). Applying machine learning techniques for detection of malicious code in network traffic. In *KI 2007: Advances in Artificial Intelligence: 30th Annual German Conference on AI, KI 2007, Osnabrück, Germany, September 10-13, 2007. Proceedings 30* (pp. 44-50). Springer Berlin Heidelberg.

Garcia-Teodoro, P., Diaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. *computers & security*, *28*(1-2), 18-28.

Highnam, K., Arulkumaran, K., Hanif, Z., & Jennings, N. R. (2021). Beth dataset: Real cybersecurity data for anomaly detection research. *TRAINING*, *763*(66.88), 8.

Highnam, K., Arulkumaran, K., Hanif, Z., & Jennings, N. R. (2021). BETH Dataset: Real Cybersecurity Data for Unsupervised Anomaly Detection Research.

Jordaney, R., Sharad, K., Dash, S.K., Wang, Z., Papini, D., Nouretdinov, I. and Cavallaro, L., 2017. Transcend: Detecting concept drift in malware classification models. In *26th USENIX security symposium (USENIX security 17)* (pp. 625-642).

Kolias, C., Kambourakis, G., Stavrou, A., & Voas, J. (2017). DDoS in the IoT: Mirai and other botnets. *Computer*, *50*(7), 80-84.

Kolosnjaji, B., Demontis, A., Biggio, B., Maiorca, D., Giacinto, G., Eckert, C. and Roli, F., 2018, September. Adversarial malware binaries: Evading deep learning for malware detection in executables. In *2018 26th European signal processing conference (EUSIPCO)* (pp. 533-537). IEEE.

Lakha, B., Mount, S. L., Serra, E., & Cuzzocrea, A. (2022, December). Anomaly Detection in Cybersecurity Events Through Graph Neural Network and Transformer Based Model: A Case Study with BETH Dataset. In *2022 IEEE International Conference on Big Data (Big Data)* (pp. 5756-5764). IEEE.

Marchetti, M., Pierazzi, F., Colajanni, M., & Guido, A. (2016). Analysis of high volumes of network traffic for advanced persistent threat detection. *Computer Networks*, *109*, 127-141.

Meng, Y. X. (2011, July). The practice on using machine learning for network anomaly intrusion detection. In *2011 International Conference on Machine Learning and Cybernetics* (Vol. 2, pp. 576-581). IEEE.

Mitchell, R., & Chen, R. (2013). Adaptive intrusion detection of malicious unmanned air vehicles using behavior rule specifications. *IEEE transactions on systems, man, and cybernetics: systems*, *44*(5), 593-604.

Nataraj, L., Karthikeyan, S., Jacob, G. and Manjunath, B.S., 2011, July. Malware images: visualization and automatic classification. In *Proceedings of the 8th international symposium on visualization for cyber security* (pp. 1-7).

Natekin, A., & Knoll, A. (2013). Gradient boosting machines, a tutorial. *Frontiers in neurorobotics*, *7*, 21.

Nevavuori, P., & Kokkonen, T. (2019). Requirements for training and evaluation dataset of network and host intrusion detection system. In *New Knowledge in Information Systems and Technologies: Volume 2* (pp. 534-546). Springer International Publishing.

Omar, L., & Ivrissimtzis, I. (2019). Using theoretical ROC curves for analysing machine learning binary classifiers. *Pattern Recognition Letters*, *128*, 447-451.

Raschka, S. (2018). Model evaluation, model selection, and algorithm selection in machine learning. *arXiv preprint arXiv:1811.12808*.

Raff, E., Zak, R., Cox, R., Sylvester, J., Yacci, P., Ward, R., Tracy, A., McLean, M. and Nicholas, C., 2018. An investigation of byte n-gram features for malware classification. *Journal of Computer Virology and Hacking Techniques*, *14*, pp.1-20.

Rieck, K., Trinius, P., Willems, C. and Holz, T., 2011. Automatic analysis of malware behavior using machine learning. *Journal of computer security*, *19*(4), pp.639-668.

Rot, A., & Olszewski, B. (2017, September). Advanced Persistent Threats Attacks in Cyberspace. Threats, Vulnerabilities, Methods of Protection. In *FedCSIS (Position Papers)* (pp. 113-117).

Saxe, J. and Berlin, K., 2015, October. Deep neural network based malware detection using two dimensional binary program features. In *2015 10th international conference on malicious and unwanted software (MALWARE)* (pp. 11-20). IEEE.

Sayin, M. O., & Başar, T. (2017, October). Secure sensor design for cyber-physical systems against advanced persistent threats. In *international conference on decision and game theory for security* (pp. 91-111). Cham: Springer International Publishing.

Schapire, R. E. (2013). Explaining adaboost. In *Empirical Inference: Festschrift in Honor of Vladimir N. Vapnik* (pp. 37-52). Berlin, Heidelberg: Springer Berlin Heidelberg.

Scherrer, C., Tewari, A., Halappanavar, M. and Haglin, D., 2012. Feature clustering for accelerating parallel coordinate descent. *Advances in Neural Information Processing Systems*, *25*.

Sgandurra, D., Muñoz-González, L., Mohsen, R. and Lupu, E.C., 2016. Automated dynamic analysis of ransomware: Benefits, limitations and use for detection. *arXiv preprint arXiv:1609.03020*.

Sharafaldin, I., Lashkari, A. H., & Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, *1*, 108-116.

Sharafaldin, I., Habibi Lashkari, A., & Ghorbani, A. A. (2019). A detailed analysis of the cicids2017 data set. In *Information Systems Security and Privacy: 4th International Conference, ICISSP 2018, Funchal-Madeira, Portugal, January 22-24, 2018, Revised Selected Papers 4* (pp. 172-188). Springer International Publishing.

Sommer, R., & Paxson, V. (2010, May). Outside the closed world: On using machine learning for network intrusion detection. In *2010 IEEE symposium on security and privacy* (pp. 305-316). IEEE.

Tankard, C. (2011). Advanced persistent threats and how to monitor and deter them. *Network security*, *2011*(8), 16-19.

Tavallaee, M., Bagheri, E., Lu, W., & Ghorbani, A. A. (2009, July). A detailed analysis of the KDD CUP 99 data set. In *2009 IEEE symposium on computational intelligence for security and defense applications* (pp. 1-6). Ieee.

Tobiyama, S., Yamaguchi, Y., Shimada, H., Ikuse, T. and Yagi, T., 2016, June. Malware detection with deep neural network using process behavior. In *2016 IEEE 40th annual computer software and applications conference (COMPSAC)* (Vol. 2, pp. 577-582). IEEE.

Ullah, I., & Mahmoud, Q. H. (2020, May). A scheme for generating a dataset for anomalous activity detection in iot networks. In *Canadian conference on artificial intelligence* (pp. 508-520). Cham: Springer International Publishing.

Ussath, M., Jaeger, D., Cheng, F., & Meinel, C. (2016, March). Advanced persistent threats: Behind the scenes. In *2016 Annual Conference on Information Science and Systems (CISS)* (pp. 181-186). IEEE.

Vigneswaran, R.K., Vinayakumar, R., Soman, K.P. and Poornachandran, P., 2018, July. Evaluating shallow and deep neural networks for network intrusion detection systems in cyber security. In *2018 9th International conference on computing, communication and networking technologies (ICCCNT)* (pp. 1-6). IEEE.

Wang, G., Cui, Y., Wang, J., Wu, L., & Hu, G. (2021). A novel method for detecting advanced persistent threat attack based on belief rule base. *Applied Sciences*, *11*(21), 9899.

Webb, G. I., Keogh, E., & Miikkulainen, R. (2010). Naïve Bayes. *Encyclopedia of machine learning*, *15*(1), 713-714.

Ye, Y., Wang, D., Li, T. and Ye, D., 2007, August. IMDS: Intelligent malware detection system. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining* (pp. 1043-1047).

Yuan, Z., Lu, Y., Wang, Z. and Xue, Y., 2014, August. Droid-sec: deep learning in android malware detection. In *Proceedings of the 2014 ACM conference on SIGCOMM* (pp. 371-372).

Zhang, Z. (2016). Introduction to machine learning: k-nearest neighbors. *Annals of translational medicine*, *4*(11).